CLAIMS:

1. A method of developing a computer software system, comprising the steps of:

   defining a first interface between a proposed view sub-system and a proposed

business logic sub-system;

   defining a second interface between a proposed handler sub-system and the proposed

business logic sub-system;

   creating the proposed view sub-system in accord with the first interface; and

   creating the proposed handler sub-system in accord with the second interface.

2. The method according to claim 1, further comprising the steps of:

   defining a third interface between the proposed view sub-system and the proposed

handler sub-system; and

   creating the proposed view sub-system in accord with both the first and third

interfaces.

3. The method according to claim 1, further comprising the steps of:

   defining a fourth interface between the proposed view sub-system and the proposed

handler sub-system; and

   creating the proposed handler sub-system in accord with both the second and the

fourth interfaces.

4. The method according to claim 1, further comprising the steps of:

   defining a third interface between the proposed view sub-system and the proposed

handler sub-system;

   defining a fourth interface between the proposed view sub-system and the proposed

handler sub-system;

creating the proposed view sub-system in accord with both the first and third interfaces; and

creating the handler sub-system in accord with both the second and the fourth interfaces.

5. The method according to claim 1, wherein:

the first interface defines a plurality of methods for data storage and retrieval that are implemented in the business logic sub-system.

6. The method according to claim 1, wherein:

the second interface defines a plurality of methods of business logic that are implemented in the business logic sub-system.

7. The method according to claim 2, wherein:

the third interface is a listener interface that defines a plurality of methods in the handler sub-system which respond to actions in the view sub-system.

8. The method according to claim 3, wherein:

the fourth interface defines a plurality of methods which are implemented in the view sub-system for use by the handler sub-system.

9. The method according to claim 1, wherein:

the view sub-system includes a plurality of user interface objects;

the handler sub-system includes a plurality of use case control objects; and

the business logic sub-system includes a plurality of business logic objects.

10. The method according claim 1, wherein:

the sub-systems are created substantially independently of each other once the interfaces have been defined.

11. A computer software system comprising:

a view sub-system including presentation objects which provide a user interface;

a business logic sub-system including use case objects which hold business data and implement business functions;

a handler sub-system including controller objects which control actions of the view sub-system and actions of the business logic sub-system;

a data interface through which the view sub-system obtains business data for the presentation objects; and

a business interface through which the handler sub-system invokes business functions.

12. The system according to claim 11, further comprising:

a listener interface through which the handler sub-system responds to events in the user interface.

12. The system according to claim 11, further comprising:

a view action interface through which the handler sub-system invokes actions in the user interface.

13.   A computer program comprising:

at least one view object including presentation objects which provide a user interface;

at least one business logic object holding business data and implementing business functions;

at least one handler object which controls actions of at least one of the view objects and actions of at least one of the business logic objects;

5   a data interface through which the at least one view object obtains business data for the presentation objects; and

a business interface through which the at least one handler object invokes business functions.

10

14. The computer program according to claim 13, further comprising:

a listener interface through which the handler object responds to events in the user interface.

15. The computer program according to claim 13, further comprising:

a view action interface through which the handler object invokes actions in the user interface.